

Name:

Student id:

Section: Serial#:

QUESTION #	1	2	3	4	5	TOTAL
MAX POINTS	15	13	8	12	20	
POINTS EARNED						

UNIVERSITY OF BAHRAIN

COLLEGE OF INFORMATION TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE

TIME: 90 MINUTES

ITCS242: ASSEMBLY LANGUAGE PROGRAMMING

SECOND TEST

DATE: DEC 19, 2013

QUESTION ONE:

{10+5 pts}

- 1) Given: num dword ?; Write the needed instructions to perform: if num is even, then store in eax the count of zeroes in num, otherwise store in eax the count of ones in num.

```
MOV     EAX, 0
MOV     ECX, 32
MOV     EBX, 0
BT      NUM, 0
JC      L2
L1: BT   NUM, EBX      ; Counting zeroes
JC      L3
INC     EAX
L3: INC  EBX
LOOP    L1
JMP    DONE
L2: BT   NUM, EBX      ; Counting ones
JNC     L4
INC     EAX
L4: INC  EBX
LOOP    L2
```

DONE:

- 2) Write the needed instructions to divide the value in esi by the value BX and store the quotient in the left half of esi and the remainder in the right half of esi.

```
SHLD    EDX, ESI, 16
SHRD    EAX, ESI, 16
DIV     BX
MOV     SI, AX
SHL     ESI, 16
MOV     SI, DX
```

Name:

Student id:

Section: Serial#:

QUESTION TWO: Write a sequence of assembly instructions to perform each of the following tasks:

- 1) Give ONE instruction to reset the even-numbered bits in EBX. Keep other bits unchanged. {2 pts}

```
AND EBX, 0AAAAAAAAH
```

- 2) Give no more than 3 instructions to copy the left half and the right halves of **eflags** into **ax** and **bx** registers correspondingly. Keep **eflags** unchanged. {3 pts}

```
PUSHFD
POP AX
pop BX
```

- 3) Give no more than 5 instructions to divide the signed predefined word values $(U1+U2) / U2$ {4 pts}

```
MOVSX EAX, U1
MOVSX EBX, U2
ADD EAX, EBX
SHLD EDX, EAX, 16
IDIV U2
```

- 4) Give no more than 5 instructions to divide the signed value in **EAX:EBX:ESI** by 16. {4 pts}

```
MOV ECX, 4
L2: SAR EAX, 1
RCR EBX, 1
RCR ESI, 1
LOOP L2
```

```
SHRD ESI, EBX, 4
SHRD EBX, EAX, 4
SAR EAX, 4
```

Name:

Student id:

Section: Serial#:

QUESTION THREE: What will be in the specified registers after executing each of the the following code segments?
{ 8*1= 8 pts}

a) MOV SP, 6C40H
POP EAX
POP SI

SP = 6C 46 H

b) MOV AX, 6C40H
MOV BX, 9E4FH
IMUL BL

AX = 13 C0 H

c) MOV AX, 79ACH
TEST AX, 8C30H
OR AX, 0F00FH

AX = F9 AF H

d) MOV AX, 9C7AH
MOV BX, 4F7CH
SHLD AX, BX, 4

AX = C7 A4 H

e) MOV AX, 3FFFFH
MOV BX, 4E50H
CMP AL, AH
JL L1
INC BL
JMP L2
L1: INC BH
L2:

BX = 4F 50 H

f) MOV AX, 3FFFFH
MOV BX, 4E50H
CMP AL, AH
JB L3
DEC BL
JMP L4
L3: DEC BH
L4:

BX = 4E 4F H

g) MOV AX, 3FFFFH
MOV BX, 6750H
AND AX, BX
ROL AX, 8

AX = 50 27 H

h) MOV AX, 3FFFFH
MOV BX, 6750H
MOV CL, 4
SAR AX, CL

AX = 03 FF H

Name:

Student id:

Section: Serial#:

QUESTION FOUR: Implement the following C++ code in Assembly language.

{12 pts}

```
int s = 0, j=40, x[40]= {10, -12, ...}; // Assume int occupies 2 bytes
while (j >= 0)
{ if (x[j] % 2 != 0)
    s += x[j];
  j--;
}
cout << s << endl;
```

```
.data
X  SWORD  10, -12, ...
S  SDWORD 0

.code
MOV     EBX, LENGTHOF X

WH: CMP     EBX, 0
    JE      EX

    BT      X[2*EBX-2], 0
    JNC     L2
    MOVSB   EAX, X[2*EBX-2]
    ADD     S, EAX

L2: DEC     EBX
    JMP     WH

EX: MOV     EAX, S
    CALL    WRITESTRING
    CALL    CRLF
```

QUESTION FIVE:

{20 pts}

Write an Assembly program that defines in the data segment two variables $x1$ and $y1$ to store values in the range -128 to $+127$. The program consists of the two procedures described as follows:

- a) The procedure **FUN** accepts 2 parameters x and y of **byte** size, calculates f as shown below, and returns the value of f in **ax**. (Write the procedure **FUN** in a form that allows invoke statement)

$$f = \begin{cases} x - y & \text{if } x \geq y \\ y - x & \text{if } x < y \end{cases}$$

- b) The procedure **main** performs the following tasks:

- 1) Prompts the user to enter from the KBD 2 values (-128 to $+127$) for variables $x1$ and $y1$.
- 2) Applies the procedure **FUN** to calculate the value of f for $x1$ and $y1$.
- 3) Displays in signed **DECIMAL** the result returned by **FUN** at the beginning of a new line.
- 4) Repeats steps from 1 to 3 40 times.

Name:

Student id:

Section: Serial#:

```
                INCLUDE    IRVINE32.INC

                .DATA
X1              sbyte     ?
Y1              sbyte     ?
SM              byte      "Enter 2 signed byte values: ",0

                .CODE
; *****
FUN            PROC        USES BX ESI, X:BYTE, Y:BYTE
                MOVSBX     AX, X
                MOVSBX     BX, Y

                CMP        AX, BX
                JL         L2

                SUB        AX, BX
                JMP        NXT

                L2: SUB     BX, AX
                MOV        AX, BX

                NXT: RET
FUN            ENDP
; *****
MAIN           PROC
                MOV        ECX, 40

                LX: LEA     EDX, SM
                CALL       WRITESTRING

                CALL       READINT
                MOV        X1, AL
                CALL       READINT
                MOV        Y1, AL
                CALL       CRLF

                INVOKE     FUN, X1, Y1

                MOVSBX     EAX, AX
                CALL       WRITEINT
                CALL       CRLF

                LOOP       LX
; *****
                EXIT
MAIN           ENDP
                END        MAIN
```